CSE8803-EPI

Using machine learning model and statistics model to forecast different targets of Covid-19

Final Project

Te-Kai Chen
Computational Science and Engineering
Georgia Institute of Technology
Atlanta GA US
tchen483@gatech.edu

Chen Chen
College of Computing
Georgia Institute of Technology
Atlanta GA US
chesterchen@gatech.edu

1 Introduction/Motivation

The Covid-19 pandemic has caused worldwide unrest and forcing local organization to measure to reduce its spread. As of October 2021, more than 4 million cases of COVID-19 have been recorded in a total of 187 countries and regions resulting in more than 250,000 deaths, while more than 1.13 million people have recovered. This final project will investigate the accuracy of a variety of machine learning model and non-machine learning model to approach for coronavirus outbreak detection in serval aspects, which including different regions (US) and counties (US) with the highest number of death cases. For purpose of forecasting the impact of the disease, we will use machine learning model and statistical model to investigate on multiple targets of covid-19 data including cases, mortality, hospitalizations, and reproduction number (Rt). In other words, this final project will concentrate on evaluating the machine learning based model and non-machine learning based model for the purpose of performance comparison. More specifically, there are specific models will be chosen for both machine learning model and non-machine learning base model to analyze and visualize their underlying behavior s in different situations. It is significant to indicate the better model describing Covid-19 and what range of data would be considered as a top priority to improve performance and accuracy.

Response to Milestone Comment

1. The plots need to be improved - everything is too small to read.

Improve the graph quality and enlarge the graph.

2. You need to do *real-time* forecasting forecast.

Add the real-time forecasting experiment for machine learning and statistic model

3. Can you try a mechanistic model as well? Like an SEIR model?

Since we comparing the machine learning model and non-machine learning model, SEIR model isn't in our model lists. However, we are willing to try a mechanistic model such as SEIR and SEIRD in the future.

2 Problem Definition

For problem definition, we set the time range for datasets that would be taken from March 2020 to June 2020 and predicate the trend to December 2020. This final project will focus on the main frame of machine learning based model and non-machine learning based model to estimate our targets. Observing the target which can be easier to forecast and evaluate the output to compare the advantage and disadvantage of each model. We will find out the best model that suitable for our dataset and indicate the performance. Moreover, we will divide US into four different regions (North-East, Mid-West, South and West) to find out the trend using both machine learning model and non-machine learning model. Besides, we will also use the five county which has the top population in the US. Last but not least, we select impatient beds used data for future prediction. For both machine learning model and non-machine learning model, this project will introduce the algorithm and proposed ideas to analyze data and try to find out the best model trained by regional data that can forecast properly for coronavirus outbreak. For the final project, we are tending to use several datasets to evaluate our model in the following circumstances. We are selecting the New York Times (NYT Times) datasets as our datasets. To

introduce the NYTimes dataset, which is releasing data with cumulative counts of coronavirus cases in the United States, this not only provides us with another form of visualization, but it also uses a formal, statistical model underneath. In other words, we can explore and find out the advantage of different models with these evaluations. Datasets contains several files that are considered as EXCEL format such as us-countiesrecent.csv, us-counties.csv, us-states.csv and us.csv. Each file includes several indexes which are date, cases, deaths, states, counties. We pick "us-states.csv", "us-counties.csv", "CDC Hospitalization Dataset by daily", "The total population (co-est2020" and "Rt_data" as history data as our target dataset in order to the next level of project research.

3 Related Work and Survey

3.1 Machine Learning based models

We are going to use four machine-learning-based models to solve the forecasting problem. We use the timeseries model to deal with this problem since our objectives are to estimate Covid-19 cases, mortalities, hospitalizations, and Rt, which are all time-related. In [9, 13], they introduced some data-driven machine learning methods that are used to perform the real-time forecasting. We then choose four models to deal with the problem. The first model is the standard Recurrent Neural Network (RNN) [1]. RNN is a model which mainly has the input of sequential data. The difference between RNN and the typical feed-forward network is that RNN will consider the current information and the previous inputs. Hence, RNN can perform better than the regular feed-forward network with this characteristic when time is a crucial factor in data. The second one is Echo State Network (ESN) [2]. The ESN model is a modification of RNN having a reservoir that randomly connects nodes inside the hidden layer. With this modification, ESN can solve some chaotic situations in the network. The third model is Long Short-Term Memory (LSTM) [14]. LSTM also has the same ideas as RNN but different from the inner blocks of the node. The LSTM block introduces the concept of gate preserving the portion of the information with current and previous information. Due to the ideas, it can help the model to learn whether the previous information is important or not. In [3], the model use LSTM to encode the data and use the attention layer to focus on the important sequence. The last model is Gated Recurrent Unit (GRU) [4]. GRU is a modified version of LSTM because it preserves the gate's idea but replaces the gates with an update gate and reset gate. This modification can help the network with fewer parameters and make the training faster and more stable. In [10], it proposed a model named MODELPREDENC using GRU as the neural network unit to encode history data to help forecast target of Covid-19. We will use the configuration in [11] to set up our models.

3.2 Non-Machine Learning based models

From Non-Machine-Learning based models, there are six different models for identifying the significant results, which including ARIMA[5], auto-regressive Integrated Moving Average[7], Hot-Winters Additive Model (HWASS)[6], TBAT, Prophet[8], and automatic Forecasting Procedure. For this project, we are going to pick three models for further investigation and find out which models can be the best to address the relationship with Covid-19. We chose several models for non-machining learning-based model and try to calibrate the parameters for each one. In other words, we select ARIMA, HWASS and Prophet for purpose of comparison. To introduce these three models, AIRMA means that we assuming a linear correlation between times series values and attempt to estimate the dependencies. One of the benefits of ARIMA can be performed in an automated way to eliminate the prediction inaccuracy. In addition, HWAAS is extend from Holt's exponential smoothing, adds the seasonality factor to the forecast, which is best for data with trend that do not fluctuate or increase. Moreover, Prophet is originally developed by Facebook to predict business-related outcomes. For purpose of forecasting, Prophet uses an easily decomposable time-series model which those models used to predict the Covid-19 cases. Besides, for training the validation, in terms of the evaluation metric, RMSE (root mean squared error) would be the indicator to each of the approaches. RMSE was employed to assess the performance of each time series model. Throughout the final project, we achieve the definition of different models and their underlying behavior.

4 Proposed Method

4.1 Machine Learning based models

We will introduce four proposed machine learning based method for our model to forecast our targets. These four models are usually used to model sequence like time-series data and nature language.

4.1.1 Recurrent Neural Network

Recurrent Neural Network (RNN) is considered the original machine learning model that use the idea "Recurrent" to model sequence data. The idea is that it performs the same task on every instance of the data which the output is related to previous sequence. To be more specific, the RNN would have the ability to preserve previous memory over computations and use the memory to help producing the output. Given a sequence $x = (x_1,$

$x_{1,\ldots,}$ x_T), the updates of the hidden state at time t, h_t, is described as below, where ϕ is a nonlinear function which can be trained by the network.

$$h_t = \begin{cases} 0, & t = 0 \\ \phi(h_{t-1}, x_t), & otherwise \end{cases}$$

Usually, the hidden state is implemented as below:

$$h_t = g(Wx + Uh_{t-1})$$

where g is the activation function such as tanh, RELU and W and U is the weighted matrix trained by the network.

4.1.2 Long Short-Term Memory (LSTM)

The existence of LSTM is because the original RNN tends to have the problem of gradient vanishing and would fail to capture long-term dependencies. The LSTM introduce three idea, which are the input gate i_t^j, forget gate f_t^j, ouput gate o_t^j and memory cell c_t^j. The LSTM not only computes h_t at each hidden state using the input and previous value of hidden state, it also computes the memory cell c_t^j at time t and unit j. The c_t^j can partially forget the existing memory using forgot gate to evaluate the importance of previous memory and adding new memory to form a new memory cell at time t. Also the output gate can modulate the amount of memory before assigning to h_t^j. We only show the equation of h_t^j and c_t^jhere, where $c_t^{\sim j}$ is the new memory content.

$$h_t^j = o_t^j tanh(c_t^j)$$

$$c_t^j = f_t^j c_{t-1}^j + i_t^j c_t^{\sim j}$$

4.1.3 Gated Recurrent Unit (GRU)

The GRU is proposed at 2014 and has the similar performance to LSTM but is computationally cheaper. In LSTM, we have to compute the memory cell at each time in each unit, needing a lot of time and space to store this weight. However, In GRU, we drop the concept of memory cell and introduce two new idea, reset gate r_t, and update gate z_t to perform the computation. These two gate is computed by the weighted matrix and input value and h_{t-1}. We first use r_t to determine the value of previous hidden unit h'_{t-1} and get h'_t in the end Secondly, we use the update gate to determine the value of current hidden state using h'_t and h_t. With this two ideas, we can reach the similar concept in LSTM with less computation. Thus, the formula is described as below.

$$h'_{t-1} = h_{t-1} \odot r_t$$

$$h'_t = tanh(Wx_t + Uh'_{t-1})$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot h'_t$$

4.1.4 Echo State Network (ESN)

The echo state network is the variation of RNN which is more efficient. The main difference between RNN is that it uses a reservoir which is a large number of sparsely connected neurons to replace recurrent layer. With the reservoir, we can reduce the computational time for processing in every neuron comparing to the original RNN. The concept of the res and the other is that it can somehow represent the chaotic situation. The most important feature is that ESN only need to train the output weight matrix W_{out} since the wight matrix of input W_{in} and reservoir W_{res} are fixed after initialization. The formula of ESN is described as below.

$$h_t = f(W_{in}u_t + W_{res}h_{t-1})$$

$$y_t = W_{out}h_t$$

where u_t ,y_t represent input and output value at time t.

4.2 Non-Machine Learning based models

For description about the approach, algorithms and models. For non-machine learning models, we use ARIMA, HWASS and Prophet to forecasting and compare which model could be the best to predicate related to COVID-19 deaths case. Also, we split into two different portions, the first one is for non-real time forecasting and the second one is for real-time forecasting, and for both portions, we apply three models (ARIMA, HWASS, Prophet) to predicate and analyze the result. For the description of algorithms it used, we used RMSE as metrics to evaluate NYT datasets, contributing with time series model HWAAS, ARIMA and Prophet. ARIMA provides a simple powerful method for making skillful time series forecasts. HWAAS is an exponential smoothing with additive trend and additive seasonality formulation. Prophet is the measure with autoregressive recurrent datasets. The RMSE is a measure of how far from the regression line data points are, which is a measure to concentrate the data is around the line of best fit. To be more detailed, we use datasets “co-est2020.csv”, “us-counties.csv”, “us-states.csv”,“state_code.csv”, “rt.csv”, “COVID 19-Reported-Patient-Impact-and Hospital-Capacity-by-State-Timeseries.csv” as our datasets. We distributed to 4 regions (e.g., North-East, Mid-West, South and West) from us-states.csv that we concentrate on the death cases, and then we apply ARIMA, HWASS and Prophet to forecast the trend and compare with the actual data. Sequentially, we take a period between March 2020 and June 2020 as 90 days history data and forecast to December 2020. Second, we use datasets “us-counties.csv” and “co-est2020.csv” to address and predict the Death cases by top five counties. Moreover, we take

"COVID-19-Reported-Patient-Impact-and-Hospital Capacity-by-State-Timeseries.csv" as datasets to analyze the impatient bed used by state. Furthermore, it would be helper for us to forecast the mean value for each region (North-East, Mid-West, South and West), then we use "rt-csv" to implement and analyze the result. In summary, we apply non-real time and real-time forecasting to analyze the trend based on 4 datasets (region, counties, impatient-bed-used, RT) and compare the accuracy and performance for those 3 models (ARIMA, HWASS, Prophet). We use RMSE as our algorithm or approach to measure the accuracy. Since RMSE is a quadratic scoring rule which measures the average magnitude of the error. Also, since the errors are squared before they are averaged, the RMSE gives a relatively high weight to large errors. Hence, we could compare each model's RMSE and determine the actual difference between each model and the actual data. This is better than the state-of-art since RMSE could be used to evaluate our statistical models and minimize the error to figure out the best performance. For example, we could use the RMSE to collect each time series model and compare different approaches in different chiromancers. Furthermore, we will indicate the results corresponding to the performance of models and find out which model would be the better option for estimate the Covid-19.

5 Experiments/Results

For Both machine learning and statistic model, we perform non-real-time forecasting and real-time forecasting on our four targets (cases, deaths, Rt and patient bed used by covid (Hp)). We will only show limited graph on this part since it'll be too messy if we put all the graph on it. The detail result is shown in the code.

5.1 Machine Learning based models

In this part, we will first preprocess our data using MinMaxscaler to scale our data into 0 to 1. This preprocessing can help us to use the model trained by regional data to predict national data without error. If we didn't preprocess the data, then it's impossible to use the regional model to predict national data due to data inconsistency.

5.1.1 Non Real-time Forecasting

The below graphs are the best model for forecasting four different targets. It's easier for model to do the non real-time forecasting since it has the previous ground truth data to help forecasting. For cases, we found out that the ESN model performs better than others. For deaths, we found out that GRU outperforms other models. For Rt, we found out that all of the model have the ability to forecast it. For patient bed usage, we found out that the GRU model has the best performance.

With the result, we can conclude that the cases is the hardest target for these models to forecast and Rt and Hp are the easier targets to forecast.

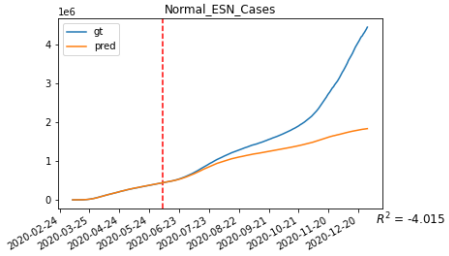


Figure 1: ESN model predicting cases

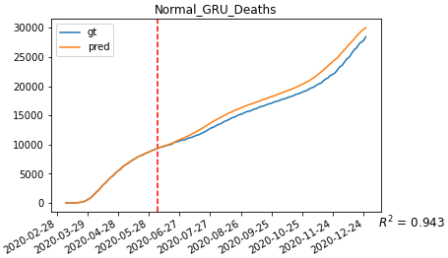


Figure 2: GRU model predicting deaths

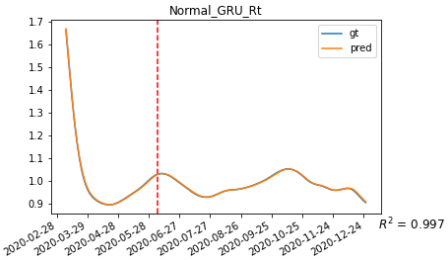


Figure 3: GRU model predicting Rt

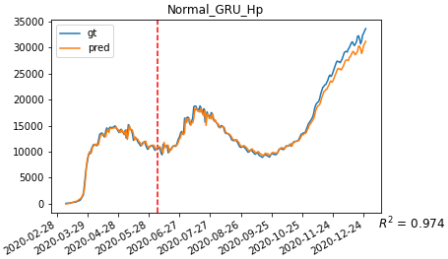


Figure 4: GRU model predicting Hp

We also try to use the models trained by regional data to forecast the national data of our four targets. Take GRU as example. After evaluating the model, we found out that the

NorthEast model has better performance on cases and the South model performs better at target death. For Rt and Hp (bed usage), seems that all model can produce good result. With the

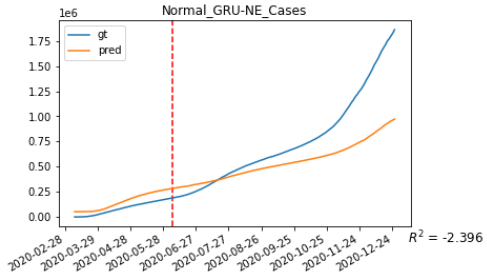


Figure 5: NorthEast-GRU model predicting cases

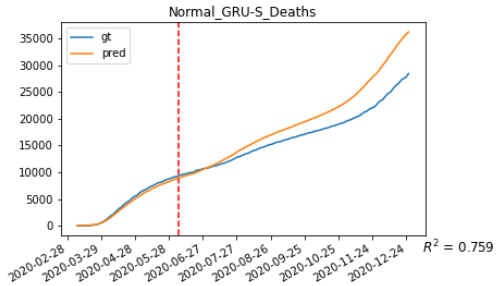


Figure 6: South-GRU model predicting deaths

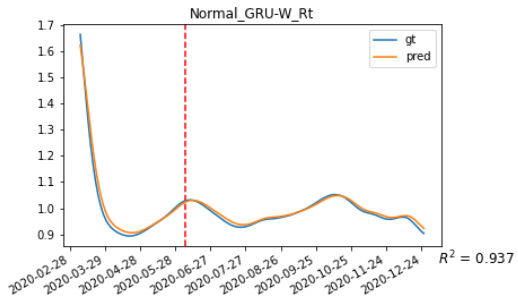


Figure 7: West-GRU model predicting Rt

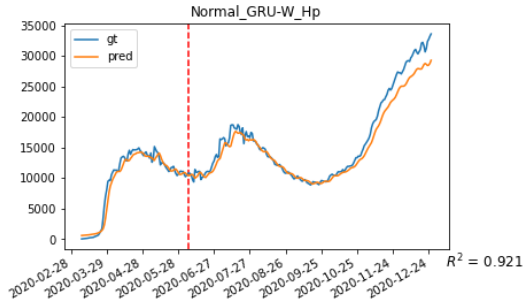


Figure 8: West-GRU model predicting Hp

5.1.2 Real-time Forecasting

In this section, we perform the real-time forecasting with four targets. However, four models fail to forecast the targets. Take RNN as example, we found out that the model fails to predict the targets when ground truth data is not given. The result of using regional model to do the real-time forecasting is pretty bad on different models either.

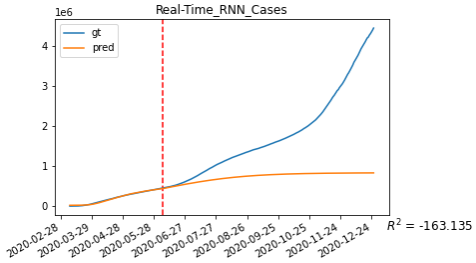


Figure 9: Real-time-RNN model predicting cases

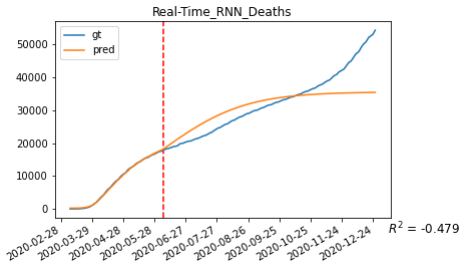


Figure 10: Real-time-RNN model predicting deaths

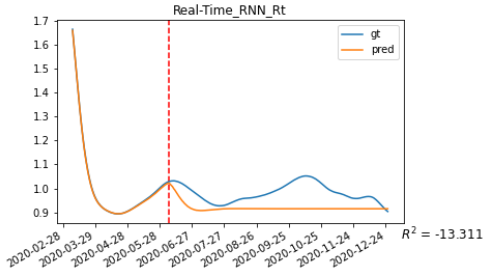


Figure 11: Real-time-RNN model predicting Rt

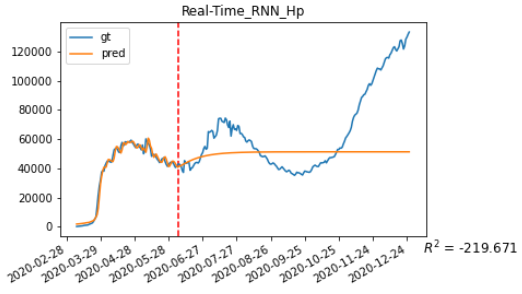


Figure 12: Real-time-RNN model predicting Hp

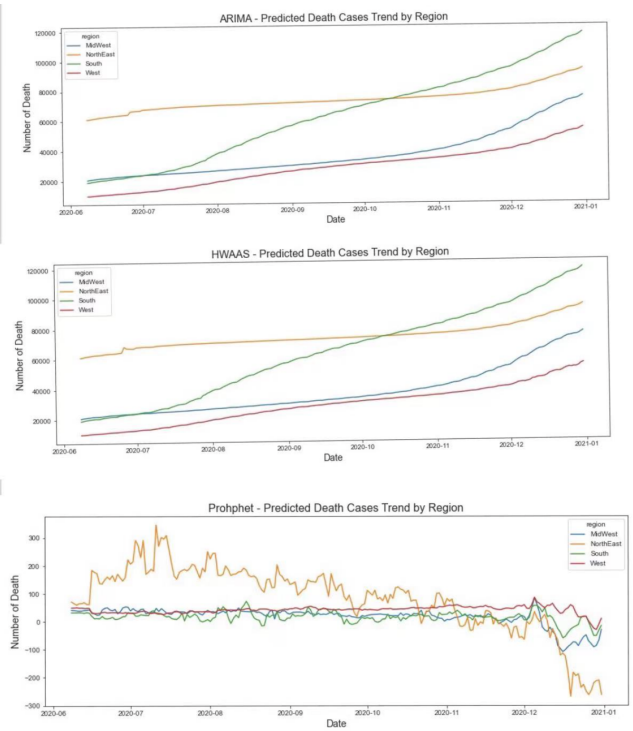
With the bad result, we think the reason is that lack of information because the model only use the previous data to predict result. However, when performing the real-time

forecasting, the given data is predicted by itself, which is not the exact trend for the real world. This would make the model lost the ability to forecast data afterward. Therefore, we need other features to help the model training like some seasonal features and related data of previous years.

In conclusion, we found that the real-time forecasting is much harder than normal forecasting without the help of ground truth data.

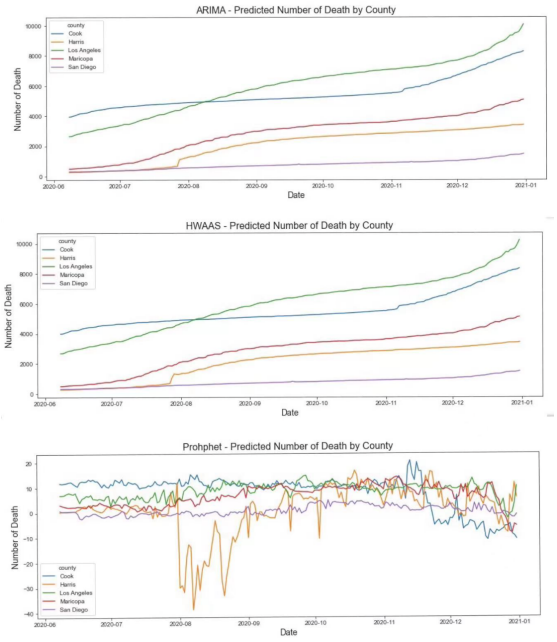
5.2 Non-Machine Learning based models

The questions need to be answered: which models among ARIMA, HWASS and Prophet can perfectly predict the Covid-19 death cases, which related to different regions, counties, impatient bed used and RT. Below is each circumnutates for the experiments of ARIMA, HWASS and Prophet. Initially for non-real time portion, we divided the state into four different regions, North-East, Mid-West, South and West. Then we can compare about among the model with regions for ARIMA, HWASS, Prophet:



Then we can compare about among the mode l with counties for ARIMA, HWASS, Prophet:

Below are the comparison (non-real time) between ARIMA, HWASS and Prophet. As we can see, we can observe that there is highly related between ARIMA and HWASS model since the trend are under similar behavior. However, Prophet has worst performance. As we can compare by the RMSE. For region, the RMSE between Mid-West, North-East, South and West, which ARIMA is: MidWest Test RMSE: 381.0805 NorthEast Test RMSE: 264.1223 South Test RMSE: 548.7780 West Test RMSE: 276.3615; For HWASS, MidWest Test RMSE: 150.3763 NorthEast Test RMSE: 198.1883 South Test RMSE: 204.8794 West Test RMSE: 120.3933. For Prophet, MidWest Test RMSE: 39418.1579 NorthEast Test RMSE: 73845.9847 South Test RMSE: 69130.2385 West Test RMSE: 30283. 5477.By similar, we have lesser RMSE value on HWASS and ARIMA, but the value is larger on Prophet. For real-time portion, we have the similar behavior that for all datasets (Counties, regions, impatient bed used), but this time we only compare the HWASS and ARIMA model. Here are graphs to compare the accuracy and performance. As we could see for regions, the RMSE for ARIMA is lesser than HWASS, which means the performance of ARIMA is better than HWASS. However, for counties comparison, RMSE in ARIMA is much larger than HWASS. In RT comparison, ARIMA is like HWASS that the accuracy and performance are similar between the two. For instance, RMSE in RT, ARIMA: MidWest Test RMSE: 1.2434 NorthEast Test RMSE: 0.8259 South Test RMSE: 2.5201 West Test RMSE: 2.8216. HWASS: MidWest Test RMSE: 13.7073 NorthEast Test RMSE: 7.9241 South Test RMSE: 1.2728 West Test RMSE: 2.6683.



6 Conclusion and Future Work

In this research experiment, we found out that real-time forecasting is much harder than normal forecasting with

ground truth data. For machine learning based model, we found out that the GRU has the better performance on the average of every targets. Also, the experiment shows that the model trained by the Northeast region has good performance on cases with non-real-time forecasting using national data and the model trained by South region can also forecast the deaths of national data. On the other hand, although we fail to perform the real-time forecasting on these targets, we still gain some insights of each model. For the future work of machine learning model, we want to find the way to deal with the problem of real-time forecasting since it's more important and more useful than forecasting with past ground truth data. Thus, to reach the better performance of real-time forecasting, we have to focus on other related and meaningful data to help us solve this problem.

For using non-machine learning model (ARIMA, HWASS, Prophet) to investigate with NYT-Covid19 data, we could observe that ARIMA has similar performance and accuracy with HWASS model, regrade to RMSE. With prediction from ARIMA and HWASS, the data predicted from models which the trend match with the actual data. Using the root mean square (RMSE) to assess the performance of each time series model, this work compares the different approaches, results indicate that, ARIMA and HWASSS overall has the better performance, and significantly better than Prophet. Future modifications to further improve the predictive accuracy of the models include the overall error as well as the time series modeling that considered other factors.

REFERENCES

[1] H. T. Siegelmann and E. D. Sontag, "Turing computability with neural nets," Applied Mathematics Letters, vol. 4, no. 6, pp. 77–80, 1991.

[2] Jaeger H, Haas H. Harnessing nonlinearity: predicting chaotic systems and saving energy in wireless communication. Science. 2004 Apr 2;304(5667):78-80.

[3] Yudistira N. COVID-19 growth prediction using multivariate long short term memory. arXiv. 2020;14(8):1–8.

[4] Chung, Junyoung, Gulcehre, Caglar, Cho, KyungHyun, and Bengio, Yoshua. Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555, 2014.

[5] Jamal Fattah, L. E. (n.d.). Forecasting of demand using ARIMA model - Jamal Fattah, Latifa Ezzine, Zineb Aman, Haj El Moussami, Abdeslam Lachhab, 2018.

[6] Yar, M., & Chatfield, C. (2002, April 23). Prediction intervals for the Holt-Winters forecasting procedure.

[7] Jebb, A. T., Tay, L., Wang, W., & Huang, Q. (2015, June 09). Time series analysis for psychological research: Examining and forecasting change.

[8] Taylor, B. S., & Letham, B. (2021, March 01). Prophet: Forecasting at scale.

[9] Siva R Venna, Amirhossein Tavanaei, Raju N Gottumukkala, Vijay V Raghavan, Anthony Maida, and Stephen Nichols. 2017. A novel data-driven model for real-time influenza forecasting. bioRxiv (2017), 185512

[10] Kamarthi, H., Rodríguez, A., & Prakash, B. A. (2021). Back2Future: Leveraging Backfill Dynamics for Improving Real-time Predictions in Future. arXiv preprint arXiv:2106.04420.

[11] Wang, Lijing, et al. "Examining Deep Learning Models with Multiple Data Sources for COVID-19 Forecasting." 2020 IEEE International Conference on Big Data (Big Data). IEEE, 2020.

[13] Hewamalage H., Bergmeir C., Bandara K. Recurrent Neural Networks for Time Series Forecasting: Current Status and Future Directions. Int.J. Forecast. 37, 388–427. (2021).

[14] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[15] Bijaya Adhikari, Xinfeng Xu, Naren Ramakrishnan, and B Aditya Prakash. 2019. Epideep: Exploiting embeddings for epidemic forecasting. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 577–586

[16] Snow, D. AtsPy: Automated Time Series Forecasting in Python. Atspy Automating Learning. (2020, April 20).

[17] Chai, Tianfeng. Root mean square error (RMSE) or mean absolute error (MAE)– Arguments against avoiding RMSE in the literature VL, 7 DO, 10.5194/gmd-7-1247-2014 JO, Geoscientific Model Development ER, AU, Draxler, R.R. PY, 2014

[18] .J.Mukherjee, M. Best research papers based on Facebook Prophet - Nerd For Tech. Time Series. Prophet Model (2021, August 11).